Operation and Installation Guide II

CLUSTERING_OPTICS

OPERATION

The program processes an image of a membrane with gold-plated particles, alongside a heavily filtered version that highlights only the particles.

It displays a clustered image of the particles to the operator.

Using the mouse, the operator can select two clusters to automatically measure the distance between their centres or measure the distance from a cluster centre to any other point on the image.

The measurements are saved to an Excel file with centre coordinates, particle count and distances. For measurements with only one centre the particle count for the second centre is -1.

Once the measurements are complete, the image is moved to a processed directory, and the program continues with the next image.

Mouse Functions:

1. **Left click**: Select two clusters, draw a line, and show the distance.
2. **Right click**: Remove the last line.
3. **Shift + Left click**: Measure distance to any point from a selected centre.
4. **Double left click**: Save measurements and move to the next image.
5. **Shift + Double left click**: Skip the image without saving.

PRECONDITIONS

The filtered files are kept in a directory, which can be at any level of the directory tree.

The original files are kept in a subdirectory to the directory and the subdirectory must be named 'original'

The filtered files have the same name as the original files, with '_filtered' added to the end of the file name

CLUSTERING

The program offers two implementations of a clustering algorithm, the basic dbscan where the epsilon is constant and the more dynamic xi, where epsilon is dynamic with an upper boundary and where the density of particles in the cluster can have a large variation.

Key Features of DBSCAN:

DBSCAN is **density-based**, meaning that it forms clusters based on areas of high point density

- **No need to specify the number of clusters**: DBSCAN automatically detects how many clusters there are based on the density of the points.
- **Can find arbitrarily shaped clusters**: DBSCAN can find clusters that are not necessarily circular or convex.
- **Robust to noise**: It can identify and label noise points, which is useful in datasets with outliers or points that don't belong to any cluster.
- **Sensitive to eps and min_samples**: The algorithm's performance is highly dependent on these two parameters. If EPS is set too large, there is a tendency to find large stretched-out clusters. If EPS is too small, there is a tendency to get too many small clusters or noise points.
- **Difficulty with varying densities**: If the dataset has regions of different densities DBSCAN might struggle to detect clusters properly because it uses a single global eps value

Key Features of OPTICS(xi):

- **Reachability Distance**: OPTICS calculates a "reachability" distance for each point, representing how far it is from its nearest core point (other points with enough nearby neighbours).
- **Varying eps**: Unlike DBSCAN, it doesn't require a single **EPS**. Based on a maximum **EPS** the OPTICS algorithm find clusters that vary in density within this range.
- **Clusters of Varying Densities**: OPTICS can find clusters where denser regions have tighter groupings, and sparser regions have larger clusters. This helps solving the problem of DBSCAN 'stretching' the clusters.

EXECUTION

The program takes up to seven parameters. Only the path is required:

- --debug: Stops after each step for debugging.
- --eps nnn: Pixel distance to cluster particles. Default is 400
- --method: How to find clusters using either xi or dbscan method.
- --xi: If using OPTICS(xi) this is the change in gradient
- --display (nnn,mmm): Size of the display window.
- --blends (alpha, beta): Blending the original and processed images.

Example:

```
python clustering_optics.py --eps 250 --debug --display (800,800) --blends (0.5,0.5) --method xi  --xi 0.05 "c:\path\to\clustering"
```

or

```
python clustering_optics.py -h
```

or

```
python clustering_optics.py --method xi --xi 0.05 .
```

xi: defines the minimum steepness on the reachability plot that OPTICS considers as a significant drop in density, which signifies a new cluster boundary.

During execution, the program creates five directories:

- **contour**: Stores images with cluster boundaries.
- **overlay**: Overlayed images for visual comparison.
- **excel**: Stores centre coordinates.
- **logs**: Last 7 days of logs.
- **processed**: Stores completed images.

Files retain their original names in each directory

INSTALLATION

The zip file includes a requirements.txt listing the required libraries.

Python version 3.12 or higher is recommended.

The program has been tested on Windows 10, 11, and Ubuntu 22.04.

1. **Navigate to the project directory** and unzip the files.
2. **Create a virtual environment**:

   ```
   python3 -m venv venv
   ```

   If it fails, you may need to install the Python venv package,

   ```
   sudo apt install python3.12-venv
   ```

3. **Activate the virtual environment**:

   ```
   source venv/bin/activate
   ```

4. **Install dependencies**:

   ```
   pip install -r requirements.txt
   ```

5. **Run the program**:

   ```
   python clustering_optics.py "c:\path\to\clustering"
   ```

6. **Deactivate the environment** (optional):

   ```
   deactivate
   ```